Functional relevance learning in generalized learning vector quantization

Kästner, Marika; Hammer, Barbara; Biehl, Michael; Villmann, Thomas

Published in:
Neurocomputing

DOI:
10.1016/j.neucom.2011.11.029

Publication date:
2012

Neurocomputing 90 (2012) 85–95

Contents lists available at SciVerse ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Functional relevance learning in generalized learning vector quantization

Marika Kästner [a], Barbara Hammer [b], Michael Biehl [c], Thomas Villmann [a,*]

[a] University of Applied Sciences Mittweida, Computational Intelligence Group, Technikumplatz 17, 09648 Mittweida, Germany
[b] University Bielefeld, Center of Excellence - Cognitive Interaction Technology CITEC, Universitätsstrasse 21-23, 33615 Bielefeld, Germany
[c] University of Groningen, Johann Bernoulli Institute for Mathematics and Computer Science, P.O. Box 407, 9700 AK Groningen, The Netherlands

ARTICLE INFO

Available online 30 March 2012

Keywords:
Functional vector quantization
Relevance learning
Feature weighting and selection
Sparse models

ABSTRACT

Relevance learning in learning vector quantization is a central paradigm for classification task depending feature weighting and selection. We propose a functional approach to relevance learning for high-dimensional functional data. For this purpose we compose the relevance profile by a superposition of only a few parametrized basis functions taking into account the functional character of the data. The number of these parameters is usually significantly smaller than the number of relevance weights in standard relevance learning, which is the number of data dimensions. Thus, instabilities in learning are avoided and an inherent regularization takes place. In addition, we discuss strategies to obtain sparse relevance models for further model optimization.

1. Introduction

Functional data frequently occur in many fields of data analysis and processing. These data usually are high-dimensional vectors representing functions with up to hundreds or thousands of dimensions [1,2]. Depending on data, the dimensions are also called (spectral) bands, time or frequency, etc. Examples for such data are time series, spectra, or density distributions to name just a few. Frequently, smoothness of the underlying function is assumed in functional data analysis [3]. The main property distinguishing functional data vectors from other high-dimensional vectors is that the sequence of the vector dimensions carries information and cannot be changed without information loss. Hence, adequate data processing is required paying attention to this special feature.

One successful method for classification of vectorial data is the robust learning vector quantization (LVQ) approach introduced by Kohonen [4]. The idea behind it is to represent the data classes by typical prototype vectors. This aim is in contrast to the widely applied support vector machines (SVM) [5,6], which cover the class borders by the so-called support vectors while maximizing the separation margin between the classes. Yet, a generalization of the standard LVQ (generalized LVQ, GLVQ) suggested by Sato and Yamada [7] has been shown to be a (hypothesis) margin optimizer [8]. Moreover, LVQ and GLVQ show robust behavior also for very high-dimensional data and are therefore suitable for functional data analysis [9].

Frequently, the different data dimensions do not equally contribute to class discrimination. Peaks or valleys in certain ranges of functional vectors may be characterizing class features and, therefore, important for classification. An automatic detection of class distinguishing vector dimensions can be incorporated into GLVQ [10]. This strategy, called *relevance learning*, can be seen as a kind of task specific metric adaptation weighting each data dimension according to its influence for class separation. The vector of all weighting parameters forms the so-called relevance profile. For the weighted Euclidean metric, the resulting generalized relevance LVQ (GRLVQ) is still a margin optimizer [11,12]. In context of high-dimensional data this relevance learning approach leads to a large number of weighting coefficients to be adapted. However, in GRLVQ each data dimension is treated independently such that for functional data the functional information would be ignored.

The idea proposed in this paper is to explicitly take into account the functional property for relevance learning in GRLVQ. In particular, the relevance profile vector is suggested to be a superposition of only a few number of basis functions. This strategy leads to a drastically reduced number of paramters to be adapted. We denote the resulting algorithm as generalized functional relevance LVQ (GFRLVQ).

The outline of the paper is as follows: First, we give a brief review of GRLVQ to clarify notations. Thereafter, we introduce the new GFRLVQ. Further, we investigate sparseness in the relevance profile of the GFRLVQ model to obtain smart models. Thereby, sparseness is considered in two different kinds: structural and feature sparseness emphasizing different aspects of relevance model reduction. The theoretical part is followed by an experimental section demonstrating

* Corresponding author.
E-mail addresses: thomas.villmann@hs-mittweida.de, villmann@hs-mittweida.de (T. Villmann).

0925-2312/$ - see front matter © 2012 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.neucom.2011.11.029

the abilities and properties of the new model compared to standard GRLVQ for different data examples.

2. Relevance learning in GLVQ—GRLVQ

To clarify notation, we start with a brief repetition of the GRLVQ stated as the standard LVQ algorithm incorporating relevance learning. It is based on GLVQ which provides a cost function for LVQ.

Generally, given a set $V \subseteq \mathbb{R}^D$ of data vectors $\mathbf{v}$ with class labels $c_{\mathbf{v}} \in Y = \{1, 2, \ldots, C\}$, the prototypes $\mathbf{w} \in W \subset \mathbb{R}^D$ with class labels y_j ($j = 1, \ldots, N$) should be distributed in such a way that they represent the data classes as accurately as possible. The following cost function, approximating the classification error, is minimized by GLVQ:

$$E(W) = \frac{1}{2} \sum_{\mathbf{v} \in V} f(\mu(\mathbf{v})) \tag{1}$$

where the function

$$\mu(\mathbf{v}) = \frac{d^+(\mathbf{v}) - d^-(\mathbf{v})}{d^+(\mathbf{v}) + d^-(\mathbf{v})} \tag{2}$$

is the classifier function with $d^+(\mathbf{v}) = d(\mathbf{v}, \mathbf{w}^+)$ denotes the dissimilarity measure between the data vector $\mathbf{v}$ and the closest prototype $\mathbf{w}^+$ with the same class label $y_{\mathbf{w}^+} = c_{\mathbf{v}}$, and $d^-(\mathbf{v}) = d(\mathbf{v}, \mathbf{w}^-)$ is the dissimilarity degree for the best matching prototype $\mathbf{w}^-$ with a class label $y_{\mathbf{w}^-}$ different from $c_{\mathbf{v}}$. The transformation function f is a monotonically non-decreasing function usually chosen as sigmoidal or the identity function. A typical sigmoidal choice is the Fermi function

$$f(x) = \frac{1}{1 + a \cdot \exp\left(-\frac{(x - x_0)^2}{2\varsigma^2}\right)} \tag{3}$$

with $x_0 = 0$ and $a = 1$ as standard parameter values. The ς-parameter allows a control of the sensitivity of the classifier at class borders but is generally fixed with $\varsigma = 1$.

The dissimilarity measure $d(\mathbf{v}, \mathbf{w})$ is supposed to be differentiable with respect to the second argument but not necessarily to be a mathematical distance. Usually, the (squared) Euclidean distance is applied. More general dissimilarity measures could also be considered. An important example is the weighted counterpart of the standard Euclidean distance

$$d^E_{\boldsymbol{\lambda}}(\mathbf{v}, \mathbf{w}) = \sum_{i=1}^{D} \lambda_i (v_i - w_i)^2 \tag{4}$$

with relevance weights $\lambda_i \geq 0$ and

$$\sum_i \lambda_i = 1 \tag{5}$$

as normalization constraint. The vector $\boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_D)^{\mathrm{T}}$ is called relevance profile.

Learning of $\mathbf{w}^+$ and $\mathbf{w}^-$ in GLVQ is done using the stochastic gradient

$$\frac{\partial_S E(W)}{\partial \mathbf{w}} = \left.\frac{\partial E(W)}{\partial \mathbf{w}}\right|_{\mathbf{v}|P(\mathbf{v})}$$

with respect to the cost function $E(W)$ for a given data vector $\mathbf{v}$ according to

$$\frac{\partial_S E(W)}{\partial \mathbf{w}^+} = \xi^+ \cdot \frac{\partial d^+(\mathbf{v})}{\partial \mathbf{w}^+} \tag{6}$$

and

$$\frac{\partial_S E(W)}{\partial \mathbf{w}^-} = \xi^- \cdot \frac{\partial d^-(\mathbf{v})}{\partial \mathbf{w}^-} \tag{7}$$

with

$$\xi^+ = f' \cdot \frac{2 \cdot d^-(\mathbf{v})}{(d^+(\mathbf{v}) + d^-(\mathbf{v}))^2} \tag{8}$$

and

$$\xi^- = -f' \cdot \frac{2 \cdot d^+(\mathbf{v})}{(d^+(\mathbf{v}) + d^-(\mathbf{v}))^2} \tag{9}$$

In general, a parametrized dissimilarity measure $d_{\boldsymbol{\lambda}}(\mathbf{v}, \mathbf{w})$ can be automatically adjusted by adaptation of the parameters, again by gradient descent:

$$\frac{\partial_S E(W)}{\partial \lambda_j} = \xi^+ \cdot \frac{\partial d^+_{\boldsymbol{\lambda}}(\mathbf{v})}{\partial \lambda_j} + \xi^- \cdot \frac{\partial d^-_{\boldsymbol{\lambda}}(\mathbf{v})}{\partial \lambda_j} \tag{10}$$

The respective algorithm is named generalized relevance LVQ—GRLVQ [10]. It should be emphasized at this point that in the GRLVQ model the relevance weights as well as the vector components are treated independently as it seems natural in the Euclidean distance or its weighted variant.

3. Functional relevance for GLVQ

As we have seen, the data dimensions are handled in GRLVQ independently according to their given (natural) ordering and, hence, an information loss occurs as a consequence in case of functional data. Further, GRLVQ requires a large number of relevance weights to be adjusted, if the data vectors are really high-dimensional as it is the case in many applications of functional data analysis. For example, processing of hyperspectral data frequently requires the consideration of hundreds or thousands of spectral bands; time series may consist of a huge number of time steps. This huge dimensionality may lead to unstable behavior of relevance learning in GRLVQ. One approach to remedy this instability is proposed by Mendenhall and Merényi suggesting a modified update strategy of GRLVQ [13], without further exploiting the functional aspect.

Therefore, relevance learning should use the functional property of data, if available, to reduce the number of parameters for relevance profile adjustment. For this purpose, we assume in the following that data vectors $\mathbf{v} = (v_1, \ldots, v_D)^T$ are representations of functions $v(t)$ with given values $v_i = v(t_i)$. Analogously we interpret the prototype vectors $\mathbf{w} = (w_1, \ldots, w_D)^T$ functionally as $w_i = w(t_i)$. Now, in *functional relevance learning* the relevance profile vector $\boldsymbol{\lambda}$ is interpreted as a function $\lambda(t)$ with $\lambda_j = \lambda(t_j)$. More precisely, the relevance function $\lambda(t)$ is supposed to be a superposition

$$\lambda(t) = \sum_{l=1}^{K} \beta_l \mathcal{K}_l(\boldsymbol{\omega}_l, t) \tag{11}$$

of K simple basis functions, $\mathcal{K}_l$ depending on only a few parameters $\boldsymbol{\omega}_l = (\omega_{l,1}, \ldots, \omega_{l,p})^\top$ and with the common restriction $\sum_{l=1}^{K} \beta_l = 1$. Doing so, each parameters β_l describes the influence of a certain basis function. The normalization constraint (5) reads now as

$$\int \lambda(t)\, dt = 1 \tag{12}$$

In consequence, the scaled Euclidean distance (4) now results in

$$d^E_{\lambda(t)}(\mathbf{v}, \mathbf{w}) = \int \lambda(t)(v(t) - w(t))^2\, dt \tag{13}$$

which can be also written as

$$d^E_{\lambda(t)}(\mathbf{v}, \mathbf{w}) = \sum_{l=1}^{K} \beta_l \int \mathcal{K}_l(\boldsymbol{\omega}_l, t)(v(t) - w(t))^2\, dt \tag{14}$$

Obviously, if the basis functions $\mathcal{K}_l(\boldsymbol{\omega}_l,t)$ form an orthogonal basis system in the Hilbert space $\mathcal{L}_2$ of quadratic integrable functions, an arbitrary approximation precision can be achieved for sufficiently large K-values [14,15]. To ensure the normalization constraint (12) the orthogonal functions have to be normalized. However, an arbitrary mixture of any kind of basis functions is possible with, maybe, reduced quality and additional requirements concerning the normalization.

Famous examples of normalized basis functions are standard Gaussians or Lorentzians:

$$\mathcal{K}_l^G(\boldsymbol{\omega}_l,t)=\frac{1}{\sigma_l\sqrt{2\pi}}\exp\left(-\frac{(t-\Theta_l)^2}{2\sigma_l^2}\right) \tag{15}$$

with $\boldsymbol{\omega}_l=(\sigma_l,\Theta_l)$ and

$$\mathcal{K}_l^L(\boldsymbol{\omega}_l,t)=\frac{1}{\eta_l\pi}\frac{\eta_l^2}{\eta_l^2+(t-\Theta_l)^2} \tag{16}$$

with $\boldsymbol{\omega}_l=(\eta_l,\Theta_l)$, respectively.

The adaptation of the relevance profile $\lambda(t)$ is now achieved by a gradient descent of the cost function with respect to the basis functions' parameters $\boldsymbol{\omega}_l$ as well as the weighting coefficients β_l. In particular, we get for the weighting coefficients of the GFRLVQ cost function

$$E_{GFRLVQ}=\frac{1}{2}\sum_{\mathbf{v}\in V}f(\mu_{\lambda(t)}(\mathbf{v})) \tag{17}$$

with classifier function

$$\mu_{\lambda(t)}(\mathbf{v})=\frac{d^+_{\lambda(t)}(\mathbf{v})-d^-_{\lambda(t)}(\mathbf{v})}{d^+_{\lambda(t)}(\mathbf{v})+d^-_{\lambda(t)}(\mathbf{v})} \tag{18}$$

the stochastic gradient as

$$\frac{\partial_S E_{GFRLVQ}}{\partial\beta_l}=\xi^+\cdot\frac{\partial d^+_{\lambda(t)}(\mathbf{v})}{\partial\beta_l}+\xi^-\cdot\frac{\partial d^-_{\lambda(t)}(\mathbf{v})}{\partial\beta_l} \tag{19}$$

with ξ^+, ξ^- as in (8) and (9), respectively. In case of the scaled Euclidean dissimilarity (14) this yields

$$\frac{\partial d^E_{\lambda(t)}}{\partial\beta_l}=\int\mathcal{K}_l(\boldsymbol{\omega}_l,t)(v(t)-w(t))^2\,dt \tag{20}$$

For the parameters $\omega_{l,i}$ the derivatives $\partial_S E/\partial\omega_{l,i}$ are calculated analogously to (19) with

$$\frac{\partial d^E_{\lambda(t)}}{\partial\omega_{l,i}}=\beta_l\int\frac{\partial\mathcal{K}_l(\boldsymbol{\omega}_l,t)}{\partial\omega_{l,i}}(v(t)-w(t))^2\,dt \tag{21}$$

for the scaled Euclidean (14), where we assumed that integration and differentiation can be interchanged. This operation is allowed if the partial derivative $|\partial\mathcal{K}_l(\boldsymbol{\omega}_l,t)/\partial\omega_{l,i}|$ can be majorized by an integrable function ϕ in the sense of Lebesgue and $\mathcal{K}_l(\boldsymbol{\omega}_l,t)$ is itself Lebesgue-integrable for each $\boldsymbol{\omega}_l$ [16]. For the introduced Gaussians $\mathcal{K}_l^G(\boldsymbol{\omega}_l,t)$ we simply obtain

$$\frac{\partial\mathcal{K}_l^G(\boldsymbol{\omega}_l,t)}{\partial\sigma_l}=\frac{1}{\sigma_l}\left(\frac{(t-\Theta_l)^2}{\sigma_l^2}-1\right)\cdot\mathcal{K}_l^G(\boldsymbol{\omega}_l,t) \tag{22}$$

and

$$\frac{\partial\mathcal{K}_l^G(\boldsymbol{\omega}_l,t)}{\partial\Theta_l}=\frac{1}{\sigma_l^2}(t-\Theta_l)\cdot\mathcal{K}_l^G(\boldsymbol{\omega}_l,t) \tag{23}$$

whereas for the Lorentzians $\mathcal{K}_l^L(\boldsymbol{\omega}_l,t)$ we have

$$\frac{\partial\mathcal{K}_l^L(\boldsymbol{\omega}_l,t)}{\partial\eta_l}=\frac{1}{\eta_l}\frac{(t-\Theta_l)^2-\eta_l^2}{\eta_l^2+(t-\Theta_l)^2}\cdot\mathcal{K}_l^L(\boldsymbol{\omega}_l,t) \tag{24}$$

and

$$\frac{\partial\mathcal{K}_l^L(\boldsymbol{\omega}_l,t)}{\partial\Theta_l}=\frac{1}{\eta_l}\frac{2(t-\Theta_l)}{\eta_l^2+(t-\Theta_l)^2}\cdot\mathcal{K}_l^L(\boldsymbol{\omega}_l,t) \tag{25}$$

The choice of basis function influences the achievable relevance profile. If smooth basis functions are applied, the resulting profile is smooth, too. Sharply peaked or non-differentiable basis functions would lead to roughly structured profiles. However, a mixture of different types is also possible to reflect different aspects. Yet, this may lead to a reduced capability in terms of minimum approximation error compared to orthogonal basis systems for fixed maximum K-value.

It should be mentioned here that we discussed so far a global metric valid for all prototypes. Obviously, this is not a real restriction: each prototype in GFRLVQ may be equipped with its own local metric, as it is also known from standard GRLVQ.

The learning of the prototypes takes place as in standard GLVQ/GRLVQ according to the formulae (6)– (9) applying the relevance profile $\lambda(t)$, i.e. replacing there the distance measure d by $d^E_{\lambda(t)}$. Thus, the learning is structurally kept as vector shifts.

4. Sparsity in GFRLVQ

One important question is the optimum number K_{opt} of basis functions needed for maximum classification performance. According to the paradigm of Occam's razor this can be related to a requirement of sparsity in the relevance model of GFRLVQ. We have to distinguish at least two different kinds of sparsity. The first one is *structural sparsity* emphasizing the sparsity of the generative model of the relevance profile with respect to the selection of number of basis functions. The second one we call *feature sparsity* reflecting the sparsity in terms of data dimensions, which are taken into account for classification, i.e. the number of t-values for which $\lambda(t)\approx 0$. Of course, feature sparsity is also be influenced by structural sparsity. However, in feature sparsity the sparseness in data dimensions used for classification is explicitly demanded.

Generally, the sparsity requirement leads to a regularization effect but needs additional terms for learning the parameters of the relevance profile $\lambda(t)$ as it is explained for both kinds of sparsity in the following subsections.

4.1. Structural sparsity

In the GFRLVQ model the number K of basis functions to be used can be chosen arbitrarily so far. Obviously, if K is too small, an appropriate relevance weighting is impossible. Otherwise, a K-value too large complicates the problem more than necessary. Hence, a good compromise is desirable. This problem can be seen as a structural sparseness requirement in functional relevance learning model. Hence, structural sparsity controls the complexity of the generating model.

A suitable methodology to judge sparsity is based on information theory. In particular, the Shannon entropy

$$H_S=-\sum_{i=1}^{K}\beta_i\log(\beta_i) \tag{26}$$

of the weighting coefficients $\boldsymbol{\beta}=(\beta_1,\ldots,\beta_K)$ can be applied to quantify structural sparsity. Maximum sparseness, i.e., minimum entropy is obtained, iff $\beta_l=1$ for exactly one l and all the other β_m are equal to zero. However, maximum sparseness may be accompanied by a decrease of accuracy in classification and/or increased cost function value E_{GFRLVQ}.

To achieve an optimal balance, we propose the following strategy: the cost function E_{GFRLVQ} from (17) is extended to

$$E^S_{GFRLVQ} = E_{GFRLVQ} + \gamma_S(\tau) \cdot H_S(\boldsymbol{\beta}) \tag{27}$$

with τ counting the adaptation steps and $\gamma_S(\tau) \geq 0$ is the weighting factor which controls the sparsity. Let τ_0 be the final time step of the GFRLVQ-learning. In the GFRLVQ with sparsity this can be interpreted such that $\gamma_S(\tau) = 0$ for $\tau < \tau_0$ holds. Thereafter, $\gamma_S(\tau)$ is slowly increased in an adiabatic manner [17], such that all parameters can persistently follow the drift of the system. An additional term for β_l-adaptation occurs for non-vanishing $\gamma_S(\tau)$-values according to this new cost function (27):

$$\frac{\partial E^S_{GFRLVQ}}{\partial \beta_l} = \frac{\partial E_{GFRLVQ}}{\partial \beta_l} + \gamma_S(\tau) \frac{\partial H_S}{\partial \beta_l} \tag{28}$$

with

$$\frac{\partial H_S}{\partial \beta_l} = -(\log(\beta_l) + 1) \tag{29}$$

This last term causes the vector $\boldsymbol{\beta}$ to become sparse. The adaptation process is stopped if the E_{GFRLVQ}-value or the classification error shows a significant increase compared to the time τ_0.

4.2. Feature sparsity

A different sparsity requirement concerns the data dimensions t contributing to the classification decision. In GFRLVQ this feature selection can be controlled by an additional additive entropy term

$$H_F(\boldsymbol{\lambda}) = -\int \lambda(t) \ln(\lambda(t))\, dt \tag{30}$$

enforcing the sparsity in the relevance profile $\lambda(t)$. This explicit sparsity control of dimensions needed for classification is substantially different from the structural sparsity. Here, the complexity of the model is not optimized but sparseness is judged in terms of non-vanishing parts of the relevance profile $\lambda(t)$. Maximum feature sparseness would lead to the so-called line spectra as known from MALDI-TOF spectra [18], for example, with spikes at the centers of the K basis functions in case of Gaussians or Lorentzians. The resulting cost function is

$$E^F_{GFRLVQ} = E_{GFRLVQ} + \gamma_F(\tau) \cdot H_F(\lambda(t)) \tag{31}$$

and the parameter $\gamma_F(\tau)$ plays the same role as $\gamma_S(\tau)$ in the case of structural sparsity.

According to the functional profile model (11) using the basis functions $\mathcal{K}_l(t)$ we obtain for the derivatives

$$\frac{\partial H_F(\boldsymbol{\lambda})}{\partial \beta_j} = -\int \mathcal{K}_j(\boldsymbol{\omega}_j, t)[1 + \ln(\lambda(t))]\, dt \tag{32}$$

and

$$\frac{\partial H_F(\boldsymbol{\lambda})}{\partial \omega_{j,i}} = -\int \beta_j \frac{\partial \mathcal{K}_j(\boldsymbol{\omega}_j, t)}{\partial \omega_{j,i}} [1 + \ln(\lambda(t))]\, dt \tag{33}$$

where we have made the same assumptions about the reversing order of integration and differentiation as before Section 3.

Obviously, the two sparsity models can be combined. However, in that case, a careful balancing between the two sparsity constraints is important to avoid instabilities and unexpected behavior.

5. Experiments and results

We tested the GFRLVQ for the classification of two well known real world spectral data sets obtained from StatLib and UCI: the *Tecator data set* [19], and the *Wine data set* [20].

The *Tecator data set* consists of 215 spectra measured for several meat probes, see Fig. 1. The spectral range is 850–1050 nm with 100 spectral bands. The data are labeled according to their fat levels into two classes (low/high). Further, the data are split randomly into 144 training and 71 test spectra.

The *Wine data set* contains 121 absorbing infrared spectra of wine between wavenumbers 4000 cm^{-1} and 400 cm^{-1} (256 bands) split into 91 training and 30 test data, see Fig. 2. These data are classified according to their two alcohol levels (low/high) as given in [21].

According to the general shape of the data, we applied GFRLVQ using Gaussian basis functions for the Tecator and Lorentzians for the Wine, because the latter one is more sharply peaked. In the first standard experiments we investigated the performance of the GFRLVQ for different numbers of prototypes and the number of basis functions to show the general capability of the algorithm and its basic behavior. In the second step we run simulation to underlay the sparsity methodology.

5.1. Standard experiments

In these experiments we varied the number of prototypes as well as the number of basis functions. In particular we used $K \in \{1, 5, 10\}$ basis functions for the experiments for each data set.

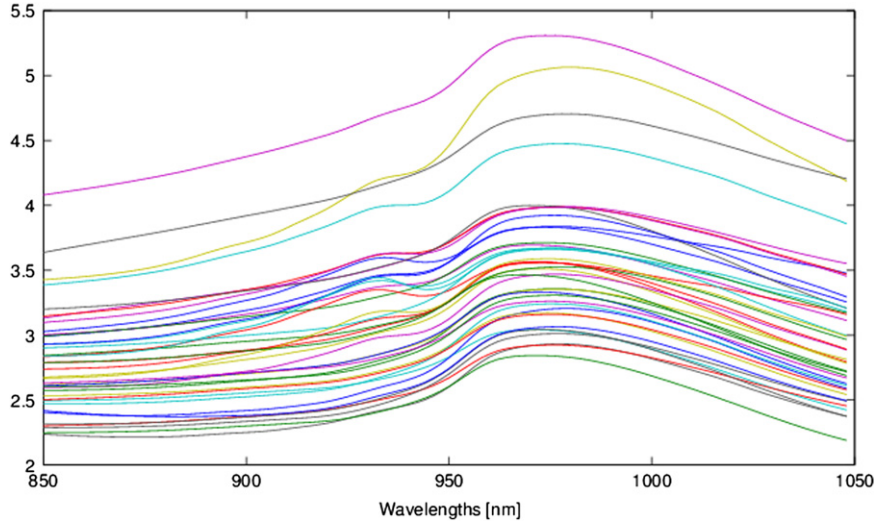

Fig. 1. The *Tecator data set*.

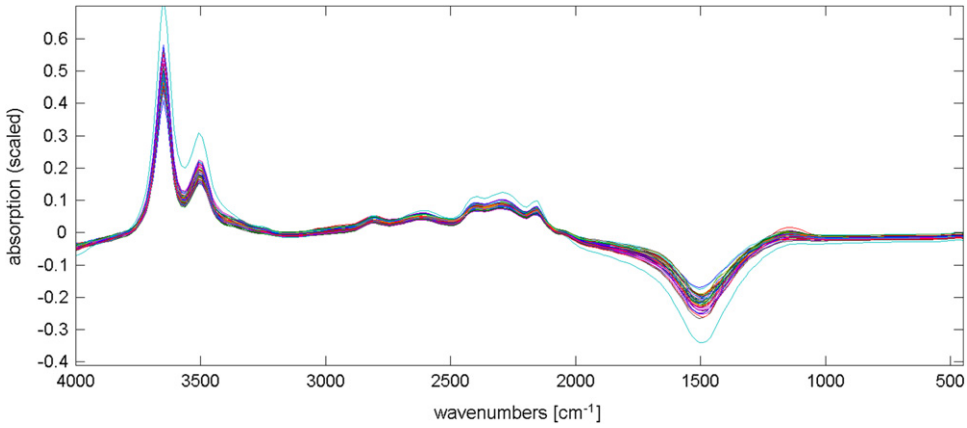

Fig. 2. The *Wine data set*.

Table 1
Correct classification rate (in %) of the training (1st value) and test (2nd value) sets with different numbers of basis functions K and prototypes $|W|$.

K	$\|W\|$					
	Tecator			Wine set		
	10	20	40	4	8	12
1	70.8 70.5	84.1 70.5	90.0 85.3	90.1 73.3	91.2 80.0	93.4 83.3
5	71.1 71.6	81.7 75.8	90.0 83.2	91.2 76.7	89.0 73.3	93.4 83.3
10	75.0 76.8	83.0 80.0	90.8 84.2	89.0 80.0	90.1 80.0	93.4 86.7
GRLVQ	71.7 70.5	87.5 77.9	94.2 83.2	93.4 83.3	91.2 86.7	93.4 80.0

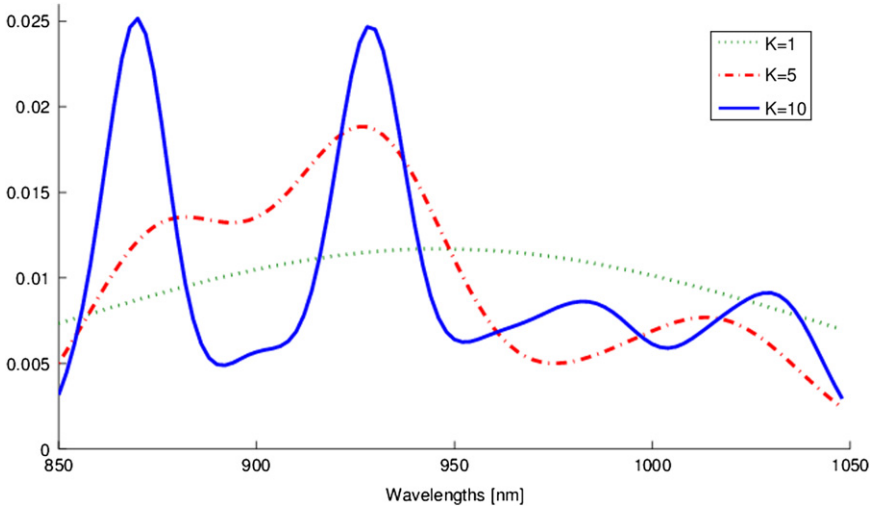

Fig. 3. Global relevance profiles obtained using different number K of Gaussian basis functions for functional relevance learning for the *Tecator data set* using 20 prototypes for learning. The richness in shape increases with K while keeping the relative smoothness.

Hence, the parameters for relevance learning are drastically reduced to 3, 15, and 30 from 100 and 256 for Tecator and Wine, respectively. The obtained classification accuracies are depicted in the Table 1. As we can see, these results are comparable to those of standard GRLVQ (see Table 1) or other approaches, see [21]. However, the results are achieved using a considerably lower number of parameters to be adapted for relevance learning compared to GRLVQ

The resulting relevance profiles for the *Tecator data set* using 20 prototypes are depicted in Fig. 3. For the case of $K=10$ Gaussian basis functions the respective decomposition is displayed in Fig. 4

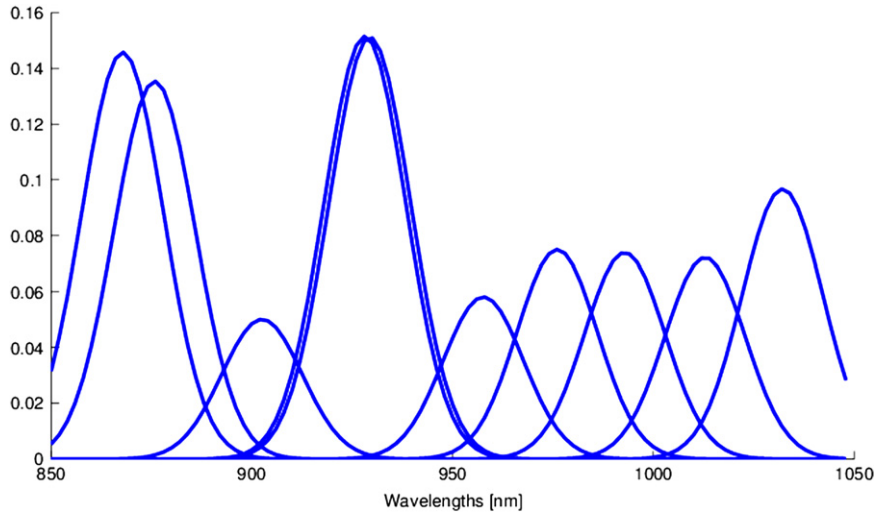

Fig. 4. Distribution of the $K=10$ adapted Gaussians for the functional relevance profile for the *Tecator data set* using 20 prototypes for learning with global metric adaptation. The resulting relevance profile is shown in Fig. 3.

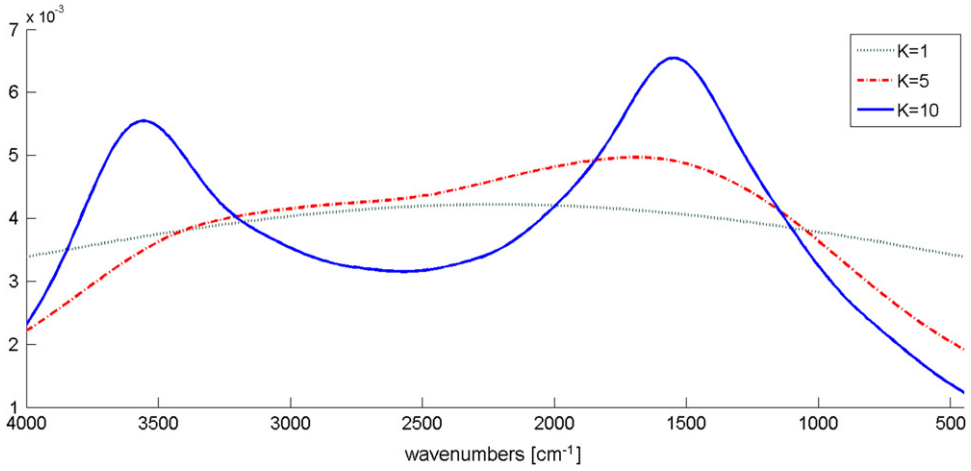

Fig. 5. Global relevance profiles obtained using different number K of Lorentzian basis functions for functional relevance learning for the *Wine data set* using 12 prototypes for learning. As for the *Tecator data set*, the richness in shape increases with K while keeping the relative smoothness.

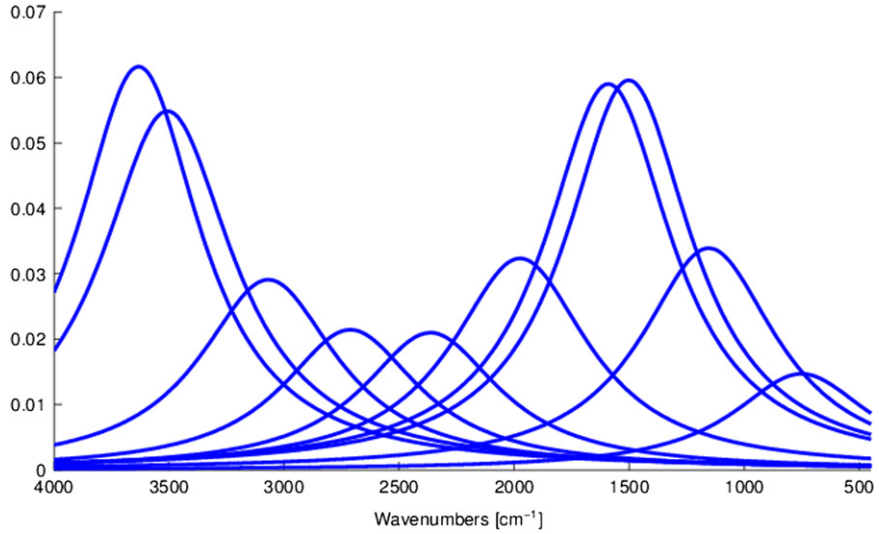

Fig. 6. Distribution of the $K=10$ adapted Lorentzians for the functional relevance profile for the *Wine data set* using eight prototypes for learning with global metric adaptation.

For the *Wine data set* the results are depicted in Figs. 5 and 6, accordingly, using eight prototypes.

As one can expect, the shape of the relevance profile becomes richer with increasing K-values, while keeping the smooth character according to the smooth basis functions applied. Further, one can observe that heights, widths as well as the centers of the basis functions were properly adapted during learning for both experiments.

5.2. Sparsity experiments

According to the introduced kinds of sparsity we performed two experiments for each data set: In the beginning, models with 20 and 12 prototypes were trained using global relevance learning and $K=10$ basis functions for each data set, respectively. There, Gaussians and Lorentzians were applied as before, accordingly. This training was based on the standard GFRLVQ cost function (17). After this basis training the influence of the sparsity constraint was slowly increased by linear growth of the weighting factor $\gamma_S(\tau)$ and $\gamma_F(\tau)$ for structural and feature sparsity, respectively.

In case of structural sparsity judged by the entropy H_S of the weighting coefficients β_l, this leads to a subsequent fading out of the several basis functions, see Fig. 7 for the *Tecator data set*. In the beginning the accuracy level is kept. Above a critical sparsity, a drastic decrease of accuracy can be observed indicating the transition from sparseness optimum to a very low level, see Fig. 8. The related relevance profiles at the beginning of the structural sparsity optimization, just before the critical transition and in the final phase, are depicted in Fig. 9. As one can see, the loss of the relevance peak around wavelength 870 nm leads to the breakdown.

For the *Wine data set* the results for structural sparsity optimization are depicted in Figs. 10–12, respectively, showing similar behavior as discussed for the *Tecator data set*. However, here the information loss immediately begins with model reduction. This is accompanied by the disappearance of the small broad plateau/peak around wavenumber 3400 cm^{-1}.

In the second experiment the feature sparsity was investigated. The experimental setting was as before for structural sparseness optimization. For the *Tecator data set* the results are depicted in Figs. 13–15. Again, we observe a drastic loss of accuracy after approximately 800 time cycles corresponding to vanishing σ_l-values for 5 basis functions. The remaining model is too poor to keep the information.

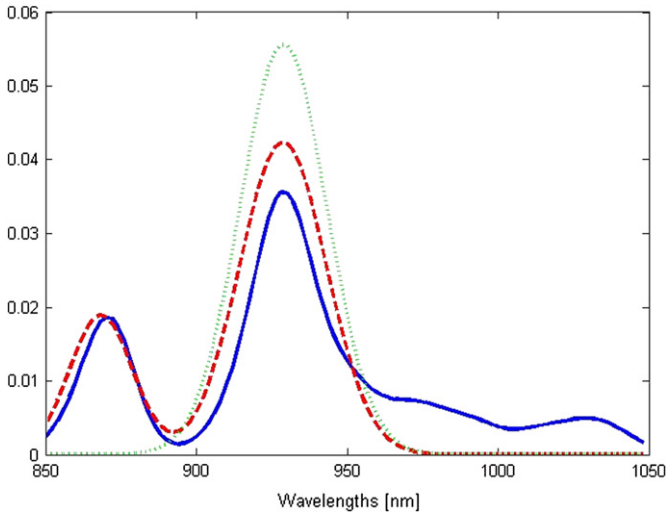

Fig. 9. Relevance profiles during structural sparseness optimization process for the *Tecator data set*: at the beginning of the structural sparsity optimization (blue—solid), just before the critical transition (red—dashed) and in the final phase (green—dotted). (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this article.)

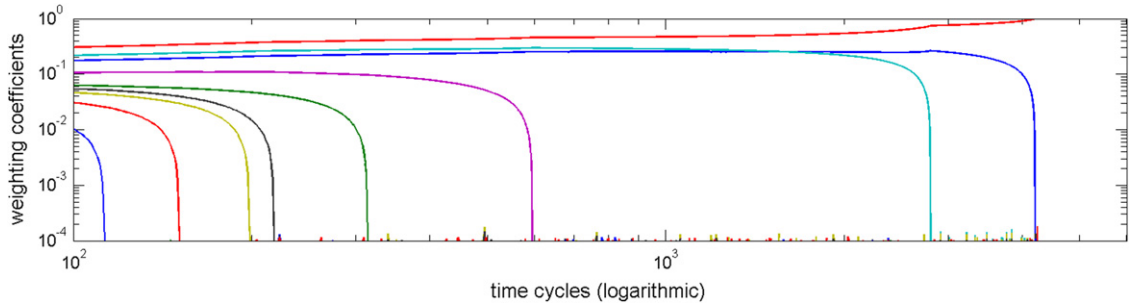

Fig. 7. Development in time of weighting coefficients β_l for structural sparsity for the *Tecator data set*. The influence of the sparsity constraint H_S was linearly increased. Global relevance learning for 20 prototypes was applied with $K=10$ Gaussians in the beginning. The weighting coefficients β_l vanish with growing sparsity pressure except only a single remaining for maximum sparseness.

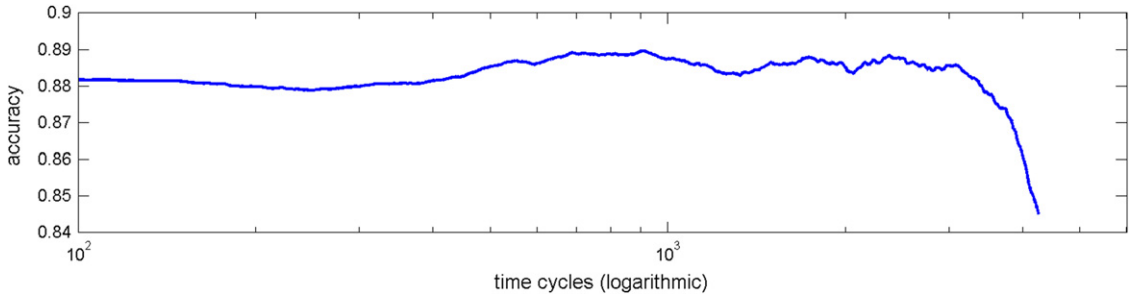

Fig. 8. Development in time of the accuracy for structural sparseness optimization for the *Tecator data set*. The accuracy is still high if at least three basis functions are weighted to be active. After one more function becomes inactive, the accuracy decreases significantly indicating a substantial information loss.

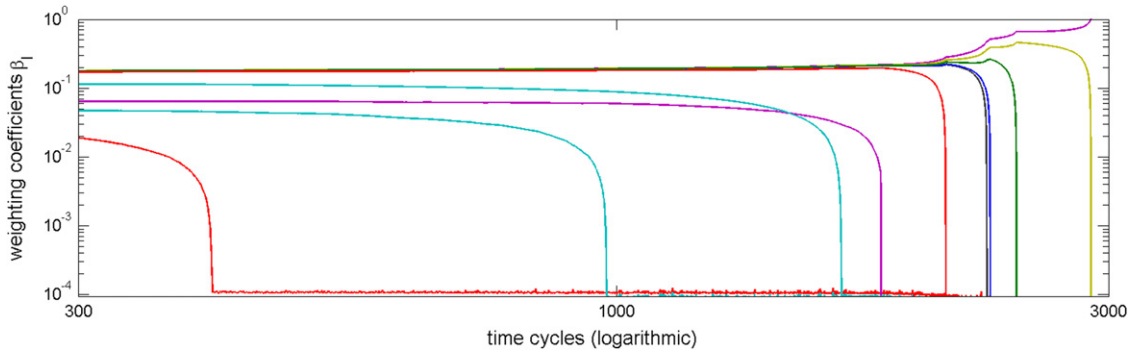

Fig. 10. Development in time of weighting coefficients β_l for structural sparsity for the *Wine data set*. The influence of the sparsity constraint H_S was linearly increased. Global relevance learning for 12 prototypes was applied with $K=10$ Lorentzians in the beginning. The weighting coefficients β_l vanish with growing sparsity pressure except only a single remaining for maximum sparseness.

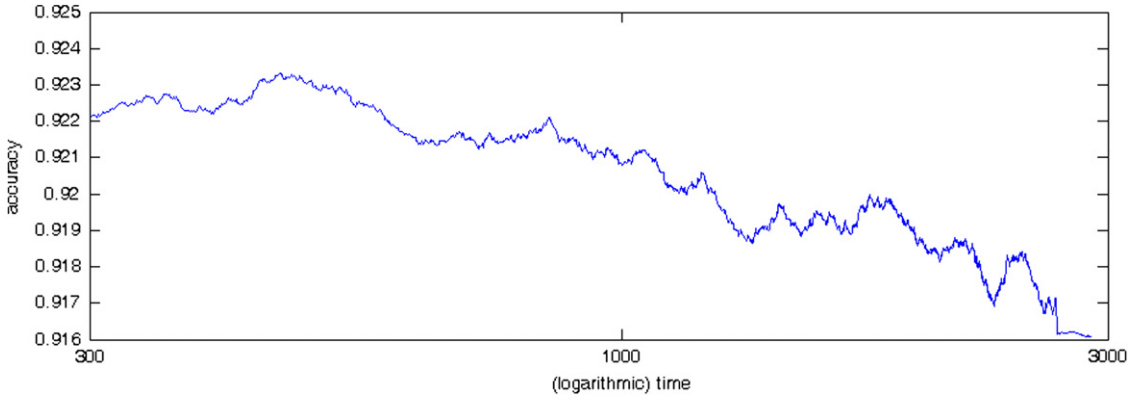

Fig. 11. Development in time of the accuracy for structural sparseness optimization for the *Wine data set*. The accuracy slowly decreases after vanishing the first basis function weight. Hence, the $K=10$ basis functions are structurally optimal.

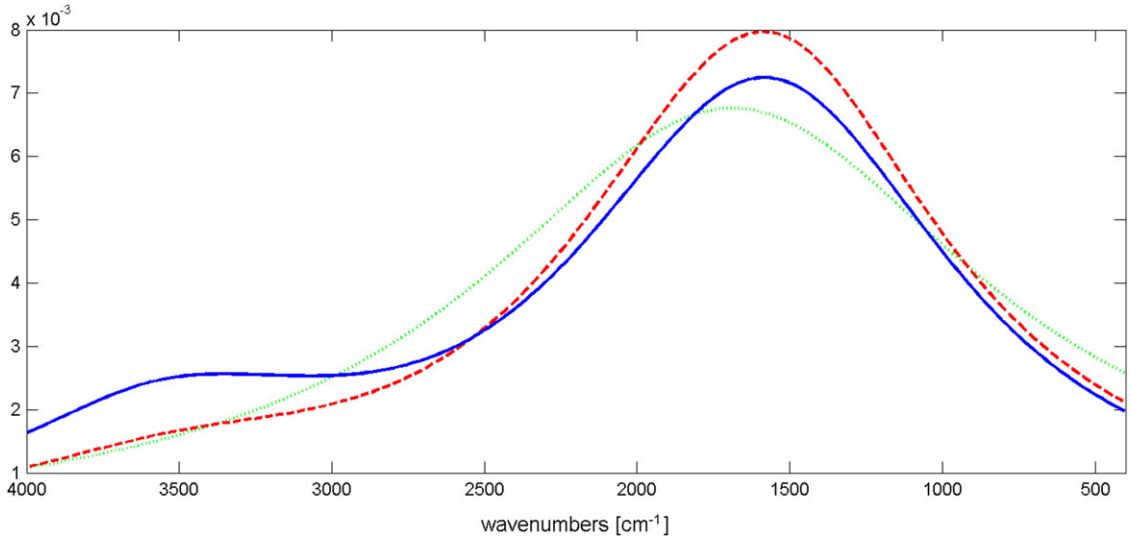

Fig. 12. Relevance profiles during structural sparseness optimization process for the *Wine data set*: at the beginning of the structural sparsity optimization (blue—solid), in the middle phase (red—dashed) and in the final phase (green—dotted). (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this article.)

For the *Wine data set* again 10 Lorentzians were taken. As one can observe in Fig. 16, the width parameters η_l immediately decreases. After vanishing of four basis functions and after 4000 learning cycles a destabilization takes places but still with keeping the high accuracy after a short stabilization phase, see Fig. 17. This process is accompanied by substantial thinning out of the relevance profile, see Fig. 18. If the pressure of the entropic term is further increased the accuracy significantly drops down and stabilizes at a lower degree leading to a very sparse relevance profile. In consequence, only a few spectral bands are sufficient to distinguish the wine alcoholic classes. This result justifies earlier findings: the influence of the data bands 130–230, corresponding to wavenumbers between 2000 cm^{-1} and 1000 cm^{-1}, seems to be class differentiating [21,22]. However, our simulations show

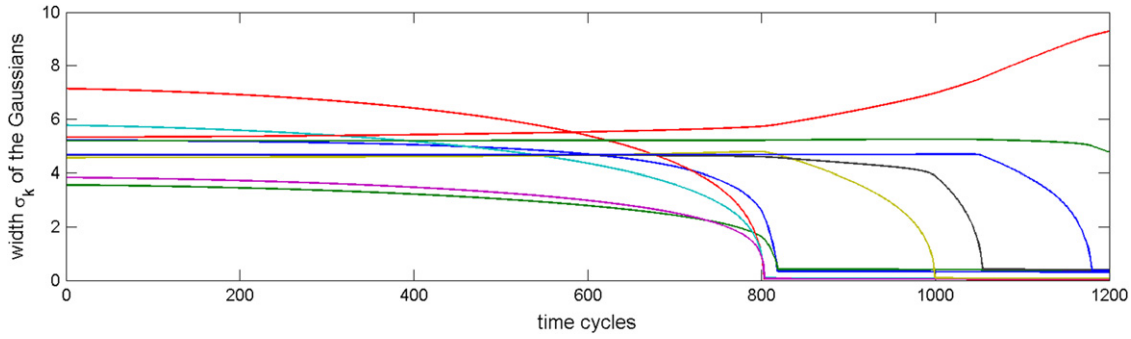

Fig. 13. Development in time of the width σ_l of the Gaussians for feature sparsity for the *Tecator data set*. The influence of the sparsity constraint H_S was linearly increased. Global relevance learning for 20 prototypes was applied with $K=10$ basis functions in the beginning. The coefficients vanish with growing sparsity pressure except only a single remaining for maximum sparseness.

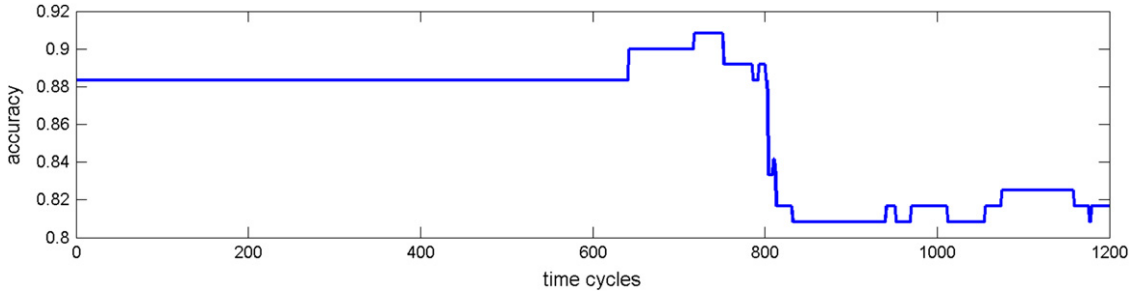

Fig. 14. Development in time of the accuracy for feature sparseness optimization for the *Tecator data set*. The accuracy is still high for at least 800 time cycles. After these, the accuracy decreases significantly indicating an substantial information loss while five basis functions vanish at this time.

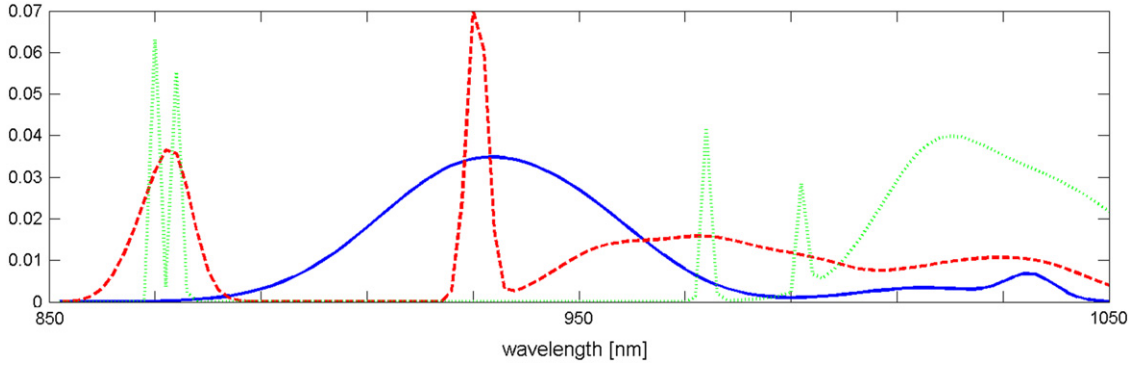

Fig. 15. Relevance profiles during the feature sparseness optimization process for the *Tecator data set*: at the beginning of the feature sparsity optimization (blue—solid), just before the critical phase transition (red—dashed) and in the final phase (green—dotted). (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this article.)

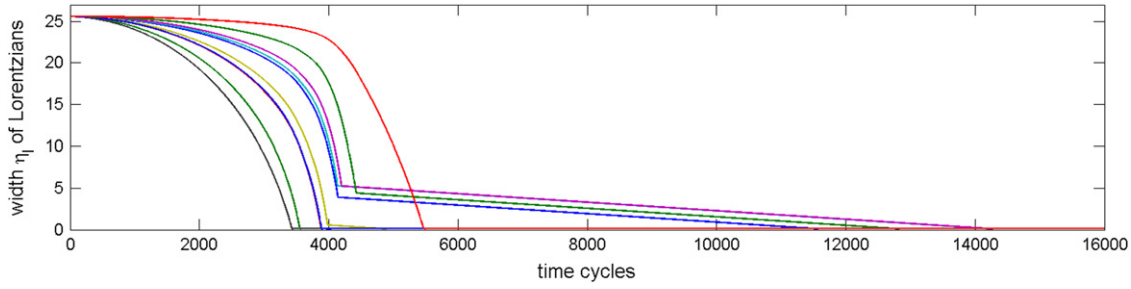

Fig. 16. Development in time of the width η_l of the Lorentzians for feature sparsity for the *Wine data set*. The influence of the sparsity constraint H_S was linearly increased. Global relevance learning for 12 prototypes was applied with $K=10$ basis functions in the beginning. The coefficients vanish with growing sparsity pressure leading to a very sparse model.

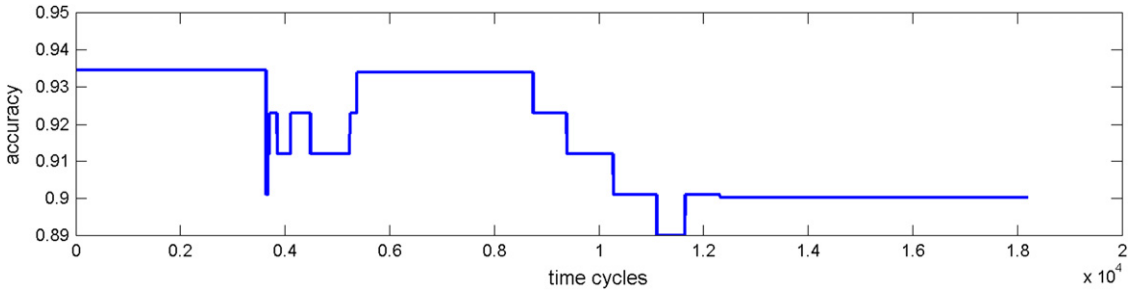

Fig. 17. Development in time of the accuracy for feature sparseness optimization for the *Wine data set*. After destabilization the accuracy significantly decreases with increasing sparseness. However, the accuracy remains still high.

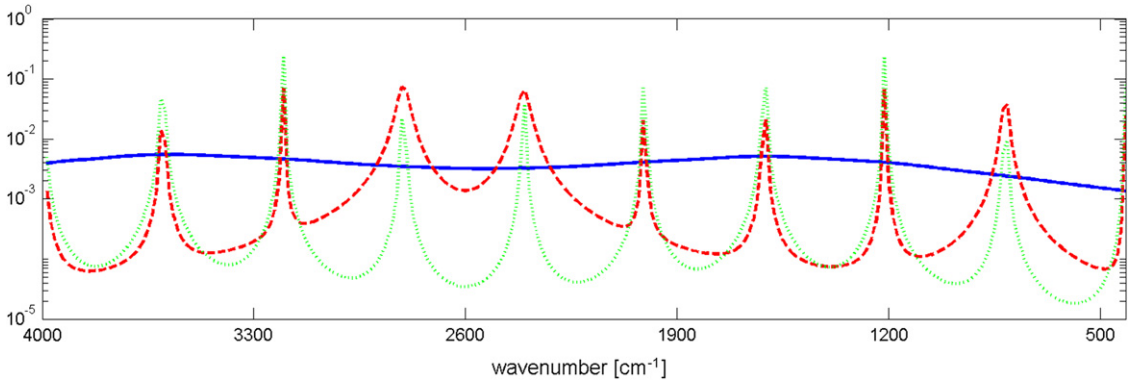

Fig. 18. Relevance profiles during the feature sparseness optimization process for the *Wine data set*: at the beginning of the feature sparsity optimization (blue—solid), just before the drop down of the accuracy after 9000 time cycles (red—dashed) and in the final phase (green—dotted). (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this article.)

that more or less the observation of single bands are sufficient and, therefore, model complexity could be drastically reduced.

6. Conclusion

In this paper we propose a *functional* relevance learning for generalized learning vector quantization. Functional learning supposes that the data vectors are representations of functions. In consequence, the relevance profiles are supposed to be functions, too, such that the profiles can be generated by a superposition of basis functions of simple shape depending on only a few parameters. Hence, the number of parameters to be adapted during functional relevance learning is drastically decreased compared to the huge number of relevance weights to be adjusted in standard relevance learning. To obtain an optimal number of basis functions for the superposition sparsity constraints are suggested dealing with different kinds of sparsity—structural and feature sparsity. The sparsity is judged in terms of the entropy of the respective sparsity model: structural sparsity prunes the superposition of the weights of the basis functions used in the model, whereas feature sparsity leads to a reduced number of input dimensions based on width adaptation of the basis functions.

We demonstrated the capabilities of the functional relevance learning algorithm for different data sets and parameter settings achieving good results compared to other models. Further, using sparsity constraints more compact models are obtained. The GFRLVQ approach is here exemplified, in the experiments for the weighted Euclidean distance based, for simplicity. Obviously, the Euclidean distance is not based on a functional norm [2,3,23]. Yet, the transfer to real functional norms and distances like Sobolev norms [24,25], the Lee-norm [23,1], kernel based LVQ-approaches [26] or divergence based similarity measures [27,28], which carry the functional aspect inherently, is straightforward and topic of future investigations.

Apparently, this functional relevance learning approach can be easily extended to matrix learning and limited rank matrix learning, which are proposed in [29,30]. This would offer a greater flexibility. First ideas are reported in [31] but are still under consideration and, therefore, subject to a forthcoming article.

References

[1] J. Lee, M. Verleysen, Nonlinear Dimensionality Reduction, Information Sciences and Statistics, Springer Science+Business Media, New York, 2007.
[2] F. Rossi, N. Delannay, B. Conan-Gueza, M. Verleysen, Representation of functional data in neural networks, Neurocomputing 64 (2005) 183–210.
[3] J. Ramsay, B. Silverman, Functional Data Analysis, 2nd ed., Springer Science+Media, New York, 2006.
[4] T. Kohonen, Self-Organizing Maps, Springer Series in Information Sciences, vol. 30, Springer, Berlin, Heidelberg, 1995 (Second Extended Edition 1997).
[5] B. Schölkopf, A. Smola, Learning with Kernels, MIT Press, 2002.
[6] J. Shawe-Taylor, N. Cristianini, Kernel Methods for Pattern Analysis and Discovery, Cambridge University Press, 2004.
[7] A. Sato, K. Yamada, Generalized learning vector quantization, in: D.S. Touretzky, M.C. Mozer, M.E. Hasselmo (Eds.), Advances in Neural Information Processing Systems 8. Proceedings of the 1995 Conference, MIT Press, Cambridge, MA, USA, 1996, pp. 423–429.
[8] K. Crammer, R. Gilad-Bachrach, A. Navot, A. Tishby, Margin analysis of the LVQ algorithm, in: S. Becker, S. Thrun, K. Obermayer (Eds.), Advances in Neural Information Processing, Proceedings of the NIPS 2002, vol. 15, MIT Press, Cambridge, MA, 2003, pp. 462–469.
[9] A. Witoelar, M. Biehl, B. Hammer, Equilibrium properties of offline LVQ, in: M. Verleysen (Ed.), Proceedings of European Symposium on Artificial Neural Networks (ESANN'2009), d-side Publications, Evere, Belgium, 2009, pp. 535–540.
[10] B. Hammer, T. Villmann, Generalized relevance learning vector quantization, Neural Networks 15 (8–9) (2002) 1059–1068.
[11] B. Hammer, M. Strickert, T. Villmann, On the generalization ability of GRLVQ networks, Neural Process. Lett. 21 (2) (2005) 109–120.
[12] B. Hammer, M. Strickert, T. Villmann, Relevance LVQ versus SVM, in: L. Rutkowski, J. Siekmann, R. Tadeusiewicz, L. Zadeh (Eds.), Artificial

Intelligence and Soft Computing (ICAISC 2004), Lecture Notes in Artificial Intelligence, vol. 3070, Springer Verlag, Berlin, Heidelberg, 2004, pp. 592–597.
[13] M. Mendenhall, E. Merényi, Relevance-based feature extraction for hyperspectral images, IEEE Trans. Neural Networks 19 (4) (2008) 658–672.
[14] A. Kolmogorov, S. Fomin, Reelle Funktionen und Funktionalanalysis, VEB Deutscher Verlag der Wissenschaften, Berlin, 1975.
[15] E. Pekalska, R. Duin, The Dissimilarity Representation for Pattern Recognition: Foundations and Applications, World Scientific, 2006.
[16] I. Kantorowitsch, G. Akilow, Funktionalanalysis in normierten Räumen, 2nd ed., Akademie-Verlag, Berlin, 1978.
[17] T. Kato, On the adiabatic theorem of quantum mechanics, J. Phys. Soc. Jpn. 5 (6) (1950) 435–439.
[18] T. Villmann, F.-M. Schleif, M. Kostrzewa, A. Walch, B. Hammer, Classification of mass-spectrometric data in clinical proteomics using learning vector quantization methods, Briefings Bioinformatics 9 (2) (2008) 129–143.
[19] TECATOR: Submitted by Hans Henrik Thodberg, Tecator meat sample dataset, available on: 〈http://lib.stat.cmu.edu/datasets/tecator〉.
[20] Wine data set provided by Prof. Marc Meurens, Available on 〈http://www.ucl.ac.be/mlg/index.php?page=databases,meurens@bnut.ucl.ac.be〉.
[21] C. Krier, M. Verleysen, F. Rossi, D. François, Supervised variable clustering for classification of NIR spectra, in: Proceedings of XVIth European Symposium on Artificial Neural Networks (ESANN 2009), Bruges, Belgique, 2009, pp. 263–268.
[22] C. Krier, F. Rossi, D. François, M. Verleysen, A data-driven functional projection approach for the selection of feature ranges in spectra with ICA or cluster analysis, Chemometrics Intelligent Lab. Syst. 91 (2008) 43–53.
[23] J. Lee, M. Verleysen, Generalization of the l_p norm for time series and its application to self-organizing maps, in: M. Cottrell (Ed.), Proceedings of Workshop on Self-Organizing Maps (WSOM) 2005, Paris, Sorbonne, 2005, pp. 733–740.
[24] B. Silverman, Smoothed functional principal components analysis by the choice of norm, Ann. Stat. 24 (1) (1996) 1–24.
[25] T. Villmann, F.-M. Schleif, Functional vector quantization by neural maps, in: J. Chanussot (Ed.), Proceedings of First Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS 2009), IEEE Press, 2009, pp. 1–4. iSBN 978-1-4244-4948-4.
[26] T. Villmann, B. Hammer, Theoretical aspects of kernel GLVQ with differentiable kernel, Ifl Technical Report Series (Ifl-09-12), 2009, pp. 133–141.
[27] T. Villmann, S. Haase, Divergence based vector quantization, Neural Comput. 23 (5) (2011) 1343–1392.
[28] E. Mwebaze, P. Schneider, F.-M. Schleif, J. Aduwo, J. Quinn, S. Haase, T. Villmann, M. Biehl, Divergence based classification in learning vector quantization, Neurocomputing 74 (9) (2011) 1429–1435.
[29] P. Schneider, B. Hammer, M. Biehl, Adaptive relevance matrices in learning vector quantization, Neural Comput. 21 (2009) 3532–3561.
[30] K. Bunte, B. Hammer, A. Wismüller, M. Biehl, Adaptive local dissimilarity measures for discriminative dimension reduction of labeled data, Neurocomputing 73 (2010) 1074–1092.
[31] M. Kästner, T. Villmann, M. Biehl, About sparsity in functional relevance learning in generalized learning vector quantization, Machine Learning Reports 5 (MLR-03-2011), 2011, pp. 1–12, ISSN:1865-3960, 〈http://www.techfak.uni-bielefeld.de/~fschleif/mlr/mlr_03_2011.pdf〉.

Marika Kästner reseived, both, her diploma in applied mathematics and her M.Sc. degrees in discrete and computer-oriented mathematics, from the University of Applied Sciences in Mittweida, Germany, in 2008 and 2010, respectively. Since 2010 she is a Ph.D.-student and the member of Computational Intelligence Group at this university.

Barbara Hammer received her Ph.D. in Computer Science in 1995 and her venia legendi in Computer Science in 2003, both from the University of Osnabrueck, Germany. From 2000 to 2004, she was leader of the junior research group 'Learning with Neural Methods on Structured Data' at the University of Osnabrueck before accepting an offer as professor for Theoretical Computer Science at Clausthal University of Technology, Germany, in 2004. Since 2010, she is holding a professorship for Theoretical Computer Science for Cognitive Systems at the CITEC cluster of excellence at Bielefeld University, Germany. Several research stays have taken her to Italy, UK, India, France, the Netherlands, and the USA. Her areas of expertise include hybrid systems, self-organizing maps, clustering, and recurrent networks as well as applications in bioinformatics, industrial process monitoring, or cognitive science.

Michael Biehl received a Ph.D. in Physics from the University of Gießen, Germany, in 1992 and the venia legendi in Theoretical Physics from the University of Würzburg, Germany, 1996. He is currently Associate Professor with Tenure in Computer Science at the Johann Bernoulli Institute, University of Groningen, The Netherlands. His main research interest is in the theory of machine learning techniques and their application in the life sciences. He is furthermore active in the modeling and simulation of complex physical system. He has co-authored more than 100 publications in international journals and conferences, preprint versions and further information can be obtained from http://www.cs.rug.nl/~biehl.

T. Villmann holds a diploma degree in Mathematics, received his Ph.D. in Computer Science in 1996 and his venia legendi in the same subject in 2005, all from University of Leipzig. From 1997 to 2009 he led the computational intelligence group of the Clinic for Psychotherapy at Leipzig University. Since 2009 he is a full Professor for Technomathematics/Computational Intelligence at the University of Applied Sciences Mittweida (Saxonia) Germany. He is founding member of the German chapter of European Neural Network Society (ENNS) and its president since 2011. His research focus includes the theory of prototype based clustering and classification, non-standard metrics, information theory and their application in pattern recognition for use in medicine, bioinformatics, remote sensing, hyperspectral analysis and others.